CLAIMS

What is claimed is:

1. A network router comprising:

queues storing data packets to be forwarded; and

a scheduler which selects queues from which packets are forwarded, the scheduler comprising:

scheduling values associated with the queues; and

a selection network by which the scheduling values are compared to select packets to be forwarded.

2. A network router as claimed in claim 1 wherein the selection network is a tree structure where each leaf of the tree structure represents a scheduling value of a queue and internal nodes of the tree structure represent winners in comparisons of scheduling values of sibling nodes of the tree structure.

3. A network router as claimed in claim 2 wherein the scheduler limits comparisons of scheduling values to a path through the tree structure from a leaf node representing a changed scheduling value to a root of the tree structure.

4. A network router as claimed in claim 2 wherein the internal nodes of the tree structure store scheduling values from winning sibling nodes.

5. A network router as claimed in claim 4 wherein the internal nodes store identities of leaf nodes corresponding to the stored scheduling values.

6. A network router as claimed in claim 2 wherein the scheduler comprises a random access memory (RAM) for storing the tree structure, an address register

which stores an address to access from the RAM a scheduling value to be compared, a compare register which stores a scheduling value to be compared to the scheduling value from the RAM and a comparator for comparing the scheduling values.

7. A network router as claimed in claim 6 wherein the scheduler further comprises hardware which receives the address in the address register and determines a sibling node where a scheduling value to be compared is stored, and determines a parent node address at which a winning compared scheduling value is stored.

8. A network router as claimed in claim 2 wherein the scheduler comprises pipeline stages, each of which compares scheduling values indicated by separate portions of the tree structure.

9. A network router as claimed in claim 8 wherein the scheduler comprises a random access memory partitioned across the pipeline stages, each partition storing at least one level of the tree structure.

10. A network router as claimed in claim 9 wherein the scheduler further comprises in each pipeline stage an address register which stores an address to access from the RAM a scheduling value to be compared, a compare register which stores a scheduling value to be compared to a scheduling value from the RAM and a comparator for comparing the scheduling values.

11. A network router as claimed in claim 2 wherein each node identifies a path to a winning leaf node.

12. A network router as claimed in claim 11 comprising a random access memory which stores leaf nodes, a flip-flop array which identifies the winner at each

internal node and a comparator for comparing scheduling values of the leaf nodes from the RAM indicated by the data stored in the flip-flop array.

13. A network router as claimed in claim 2 further comprising an indicator associated with each queue to disable the queue from scheduling.

14. A network router as claimed in claim 2 wherein the scheduling values include scheduled transmission times according to a constant-bit-rate (CBR) service guarantee.

15. A network router as claimed in claim 14 wherein the scheduling values are updated to reflect variable packet links.

16. A network router as claimed in claim 14 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

17. A network router as claimed in claim 14 further comprising scheduling values which represent theoretical transmission times using a weighted-fair-queuing (WFQ) scheduling policy.

18. A network router as claimed in claim 17 wherein the WFQ scheduling values are updated for variable packet lengths.

19. A network router as claimed in claim 17 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

20. A network router as claimed in claim 2 further comprising scheduling values which represent theoretical transmission times using a weighted-fair-queuing (WFQ) scheduling policy.

21. A network router as claimed in claim 20 wherein the WFQ scheduling values are updated for variable packet lengths.

22. A network router as claimed in claim 20 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

23. A network router as claimed in claim 1 wherein the selection network is a sorting network by which the scheduling values are compared to order the queues by scheduling priority.

24. A network router comprising:

queues storing data packets to be forwarded; and

a scheduler which selects queues from which packets are forwarded, the scheduler comprising:

scheduling values associated with the queues;

indicators associated with the queues to disable the queues; and

a comparator which compares scheduling values of queues which are not disabled to forward data packets therefrom.

25. A network router comprising:

queues storing data packets to be forwarded; and

a scheduler which selects queues from which packets are forwarded, the scheduler comprising:

first scheduling values corresponding to a first scheduling method associated with a first subset of queues;

second scheduling values corresponding to a second scheduling method associated with a second subset of queues, at least one queue being a member of each of the first subset and second subset of queues; and

a queue selector by which first scheduling values are compared and second scheduling values are compared to select packets to be forwarded.

26. A network router as claimed in claim 25 wherein the first scheduling method is constant bit rate (CBR) scheduling and the second scheduling method is weighted-fair-queuing (WFQ) scheduling.

27. A network router as claimed in claim 26 wherein the scheduler selects a queue by:

identifying an earliest scheduled CBR queue;

if the scheduling value of the identified CBR queue is less than or equal to a current time, transmitting a corresponding packet from the CBR queue and updating the CBR scheduling value associated with the queue; and

otherwise, transmitting a packet from a WFQ queue having an earliest scheduling value and updating the scheduling value of that queue.

28. A network router comprising:

queues storing data packets to be forwarded; and

a scheduler which selects queues from which packets are forwarded, the scheduler comprising:

scheduling values associated with the queues;

a selector by which scheduling values are compared to select packets to be forwarded; and

a scheduling value updater which updates the scheduling value of a queue based on a variable length of a packet in the queue.

29. A network router as claimed in claim 28 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

30. A network router as claimed in claim 29 wherein the scheduler selects a queue by:

identifying an earliest scheduled CBR queue;

if the scheduling value of the identified CBR queue is less than or equal to a current time, transmitting a corresponding packet from the CBR queue and updating the CBR scheduling value associated with the queue; and

otherwise, transmitting a packet from a WFQ packet having an earliest scheduling value and updating the scheduling value of that queue.

31. A network router as claimed in claim 28 wherein the scheduler comprises:

identifying an earliest scheduled CBR queue;

if the scheduling value of the identified CBR queue is less than or equal to a current time, transmitting a corresponding packet from the CBR queue and updating the CBR scheduling value associated with the queue; and

otherwise, transmitting a packet from a WFQ packet having an earliest scheduling value and updating the scheduling value of that queue.

32. A network router comprising:

a first set of queues storing data packets to be forwarded;

a first scheduler which selects queues of the first set of queues from which packets are forwarded to a first intermediate queue;

a second set of queues storing data packets to be forwarded;

a second scheduler which selects queues of the second set of queues from which packets are forwarded to a second intermediate queue; and

a further scheduler which selects intermediate queues from which packets are forwarded.

33. A network router as claimed in claim 32 wherein the first scheduler selects queues according to plural scheduling methods.

34. A method of routing data packets comprising:

storing data packets in queues;

associating scheduling values with the queues; and

comparing scheduling values in a selection network to select queues from which packets are forwarded.

35. A method as claimed in claim 34 wherein the selection network is a tree structure where each leaf of the tree structure represents a scheduling value of a queue and internal nodes of the tree structure represent winners in comparisons of scheduling values of sibling nodes of the tree structure.

36. A method as claimed in claim 35 wherein the scheduler limits comparisons of scheduling values to a path through the tree structure from a leaf node representing a changed scheduling value to a root of the tree structure.

37. A method as claimed in claim 35 wherein the internal nodes of the tree structure store scheduling values from winning sibling nodes.

38. A method as claimed in claim 37 wherein the internal nodes store identities of leaf nodes corresponding to the stored scheduling values.

39. A method as claimed in claim 35 wherein the tree structure is stored in a random access memory (RAM) and scheduling values from a compare register and from the RAM are compared.

40. A method as claimed in claim 39 further comprising determining a sibling node where a scheduling value to be compared is stored, and determining a parent node address at which a winning compared scheduling value is stored.

41. A method as claimed in claim 35 further comprising comparing scheduling values indicated by separate portions of the tree structure in pipeline stages.

42. A method as claimed in claim 41 further comprising storing at least one level of the tree structure in a partition of a random access memory (RAM) partitioned across the pipeline stages.

43. A method as claimed in claim 42 further comprising, in each pipeline stage, comparing scheduling values from a compare register and from the RAM.

44. A method as claimed in claim 35 wherein each node identifies a path to a winning leaf node.

45. A method as claimed in claim 44 wherein leaf nodes of the tree structure are stored in a random access memory and the winner at each internal node is identified in a flip-flip array, the method comprising comparing scheduling values of the leaf nodes from the RAM indicated by the data stored in the flip-flop array.

46. A method as claimed in claim 34 further comprising providing an indicator associated with each queue to disable the queue from scheduling.

47. A method as claimed in claim 34 wherein the scheduling values include scheduled transmission times according to a constant-bit-rate (CBR) service guarantee.

48. A method as claimed in claim 47 wherein the scheduling values are updated to reflect variable packet lengths.

49. A method as claimed in claim 47 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

50. A method as claimed in claim 47 wherein scheduling values represent theoretical transmission times using a weighted-fair-queuing (WFQ) scheduling policy.

51. A method as claimed in claim 50 wherein the WFQ scheduling values are updated for variable packet lengths.

52. A method as claimed in claim 50 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

53. A method as claimed in claim 34 wherein scheduling values represent theoretical transmission times using a weighted-fair-queuing (WFQ) scheduling policy.

54. A method as claimed in claim 53 wherein the WFQ scheduling values are updated for variable packet lengths.

55. A method as claimed in claim 53 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

56. A method as claimed in claim 34 wherein the selection network is a sorting network by which the scheduling values are compared to order the queues by scheduling priority.

57. A method of routing data packets comprising:

storing data packets in queues;

associating scheduling values with the queues;

associating indicators with the queues to disable the queues; and

comparing scheduling values of queues which are not disabled before the data packets therefrom.

58. A method of routing data packets comprising:

storing data packets in queues;

associating scheduling values corresponding to a first scheduling method with a first subset of queues;

associating scheduling values corresponding to a second scheduling method with a second subset of queues, at least one queue being a member of each of the first subset and second subset of queues; and

comparing scheduling values to select packets to be forwarded, excess capacity under the first scheduling method being available for scheduling under the second scheduling method.

59. A network router as claimed in claim 58 wherein the first scheduling method is constant bit rate (CBR) scheduling and the second scheduling method is weighted-fair-queuing (WFQ) scheduling.

60. A network router as claimed in claim 58 wherein the scheduler selects a queue by:

identifying an earliest scheduled CBR queue;

if the scheduling value of the identified CBR queue is less than or equal to a current time, transmitting a corresponding packet from the CBR queue and updating the CBR scheduling value associated with the queue; and

otherwise, transmitting a packet from a WFQ queue having an earliest scheduling value and updating the scheduling value of that queue.

61. A method of routing data packets comprising:

storing data packets in queues;

associating scheduling values with the queues;

comparing scheduling values to select data packets to be forwarded; and

updating the scheduling value of a queue based on a variable length of a packet in the queue.

62. A network router as claimed in claim 61 wherein the scheduling values are updated to reflect byte stuffing applied to a prior packet.

63. A network router as claimed in claim 62 wherein the scheduler selects a queue by:

identifying an earliest scheduled CBR queue;

if the scheduling value of the identified CBR queue is less than or equal to a current time, transmitting a corresponding packet from the CBR queue and updating the CBR scheduling value associated with the queue; and

otherwise, transmitting a packet from a WFQ packet having an earliest scheduling value and updating the scheduling value of that queue.

64. A network router as claimed in claim 61 wherein the scheduler comprises:

identifying an earliest scheduled CBR queue;

if the scheduling value of the identified CBR queue is less than or equal to a current time, transmitting a corresponding packet from the CBR queue and updating the CBR scheduling value associated with the queue; and

otherwise, transmitting a packet from a WFQ packet having an earliest scheduling value and updating the scheduling value of that queue.

65. A method of routing data packets comprising:

storing data packets to be forwarded in first and second sets of queues;

selecting queues of the first set of queues from which packets are forwarded to a first intermediate queue;

selecting queues of the second set of queues from which packets are forwarded to a second intermediate queue; and

selecting intermediate queues from which packets are forwarded.

66. A method as claimed in claim 65 wherein the step of selecting queues of the first set of queues comprises selecting queues according to plural scheduling methods.

67. A network router comprising:

queues storing data packets to be forwarded; and

scheduling means for selecting queues from which packets are forwarded the scheduling means comprising:

scheduling values associated with the queues; and

a selection network by which the scheduling values are compared to select packets to be forwarded.

68. A network router comprising:

queues storing data packets to be forwarded; and

scheduling means for selecting queues from which packets are forwarded, the scheduling means comprising:

scheduling values associated with the queues;

indicating means associated with the queues for disabling the queues; and

comparator means for comparing scheduling values of queues which are not disabled to forward data packets therefrom.

69. A network router comprising:

queues storing data packets to be forwarded; and

scheduling means for selecting queues from which packets are forwarded, the scheduling means comprising:

first scheduling values corresponding to a first scheduling method associated with a first subset of queues;

second scheduling values corresponding to a second scheduling method associated with a second subset of queues, at least one queue being a member of each of the first subset and second subset of queues; and

queue selecting means for comparing first scheduling values and second scheduling values to select packets to be forwarded.

70. A network router comprising:

queues storing data packets to be forwarded; and

scheduling means for selecting queues from which packets are forwarded, the scheduling means comprising:

scheduling values associated with the queues;

selecting means for comparing scheduling values to select packets to be forwarded; and

updating means for updating the scheduling value of a queue based on a variable length of a packet in the queue.

71. A network router comprising:

a first set of queues storing data packets to be forwarded;

first scheduling means for selecting queues of the first set of queues from which packets are forwarded to a first intermediate queue;

a second set of queues storing data packets to be forwarded;

second scheduling means for selecting queues of the second set of queues from which packets are forwarded to a second intermediate queue; and

further scheduling means for selecting intermediate queues from which packets are forwarded.